PATENT

AT9-98-544

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of:
Balaram Sinharoy

Serial No.: 09/435,070

Filed: November 4, 1999

Title: CIRCUITS, SYSTEMS AND METHODS FOR PERFORMING BRANCH PREDICTIONS BY SELECTIVELY ACCESSING BIMODAL AND FETCH-BASED BRANCH HISTORY TABLES

Before the Examiner:
William Wood

Group Art Unit: 2124

IBM Corporation
IP Law Department
Internal Zip 4054
11400 Burnet Road
Austin, Texas 78758

APPLICANT'S BRIEF IN SUPPORT OF APPEAL

Box AF
Assistant Commissioner for Patents
Washington, D.C. 20231

Dear Sir:

This brief is being submitted pursuant to 37 C.F.R. § 1.192. Appellant is furnishing herewith three (3) copies of this brief.

CERTIFICATION UNDER 37 C.F.R. § 1.8

I hereby certify that this correspondence (along with any item referred to as being attached hereto) is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to Box AF, Assistant Commissioner for Patents, Washington, D.C. 20231, on March 6, 2003.

Toni Stanley
Signature

Toni Stanley
(Printed name of person certifying)

I. REAL PARTY-IN-INTEREST

The real party-in-interest is International Business Machines Corp., who is the assignee of the entire right and interest in the present Application.

II. RELATED APPEALS AND INTERFERENCES

There are no appeals or interferences known to Appellant, the Appellant's legal representative, or assignee which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. STATUS OF CLAIMS

Claims 10 and 21-40 are pending in the Application.

Claims 1-9 and 11-20 have been cancelled without prejudice or disclaimer.

Claims 10 and 21-40 stand rejected, and are the subject of the instant appeal.

IV. STATUS OF AMENDMENTS

No amendment has been filed subsequent to final rejection.

V. SUMMARY OF THE INVENTION

A branch prediction apparatus in accordance with a preferred embodiment of the present inventive concepts includes three tables are used for branch prediction, namely, a local branch history table (LBHT), a fetch-based branch history table (GBHT) and a selector table (GSEL). (Detailed Description, page 9, lines 13-18.) In the illustrated embodiment, each table is composed of a preselected number, *n*, of entries each of which includes a number, *p*, of 1-bit counters. (Detailed Description, page 9, lines 18-20.) For purposes of the present discussion, when a given one of the counters in the local or fetch-based branch history tables is set to a Logic 1, that counter (entry) will be considered as being set to a taken branch prediction value. (Detailed Description, page 9, lines

20-22.) Consequently, in this configuration, a counter (entry) storing a Logic 0 will be considered as storing a branch not-taken-prediction bit. (Detailed Description, page 9, line 22 through page 10, line 1.) It should be noted however, that in alternate embodiments, the reverse logic may be used without deviating from the inventive concepts. (Detailed Description, page 10, lines 1-3.)

The local branch history table is accessed for obtaining branch predictions using a pointer constituting *n* bits taken from the current cache line address, in instruction fetch address register (IFAR) in a conventional fashion. (Detailed Description, page 10, lines 4-6.) Fetch-based branch history table is accessed for obtaining branch predictions in a fashion in which *n* number of bits taken from the current cache line address are bitwise XORed with *n* number of bits from GHV register. (Detailed Description, page 10, lines 6-10.)

In the preferred embodiment, the entries of GSEL are accessed for obtaining predictions using the same pointer generated for the fetch-based branch history table. (Detailed Description, page 10, lines 13-15.) T he accessed entry from selector table is then used by selection logic to select either the local branch prediction values output from LBHT or the fetch-based branch prediction value accessed from GBHT for use as the final branch prediction value for determining if the branch is to be taken or not taken. (Detailed Description, page 10, lines 15-19.) Note that a number *q* of the prediction values may be from LBHT and a remaining number *p-q* may be from GBHT. (Detailed Description, page 10, lines 21-22.) Thus, the number of predictions in an entry accommodates all of the instructions that are fetched in a single cycle, which may be referred to as a fetch group (FG). (Detailed Description, page 11, lines 1-3.) The number, *p*, of instructions in a fetch group may be eight in an embodiment of the present invention. (Detailed Description, page 11, lines 1-3.)

The GHV tracks the history of branch instructions as they are fetched and executed. (Detailed Description, page 11, lines 10-11.) Thus, as branches are executed and resolved, the GHV is updated. (Detailed Description, page 11, lines 11-12.)

Additionally, the entries in the LBHT, GBHT and GSEL 30 must also be updated in response to the execution of branch instructions. (Detailed Description, page 11, lines 15-16.) The entries are updated by providing information to the appropriate entry in the LBHT, GBHT and GSEL for setting or resetting, as appropriate, the *p* one-bit counters in the corresponding entry,

depending on the prediction and the resolution, or actual outcome, of the branch. (Detailed Description, page 11, lines 16-20.)

In this way, branch prediction based on a prediction history is implemented having a constant amount of processing. (Detailed Description, page 28, lines 15-16.) According to the principles of the present invention, one bit is shifted into the global history vector for each fetch group. (Detailed Description, page 28, lines 16-17.) The loading of a bit, "one" or a "zero," in the global history vector essentially captures the path the program has taken to reach the branch instruction being predicted, and thereby provides an indication of how the branch will behave (taken or not-taken). (Detailed Description, page 28, lines 16-17.)

VI. ISSUES

(A) Are claims 21, 22, 24, 25, 26, 28, 29, 31-36 and 38 properly rejected under 35 U.S.C. § 102 as being anticipated by *Patt et al.*, "Alternative Implementations of Hybrid Branch Predictors," Proceedings of the 28th Annual Symposium on Microarchitecture, 1995, pages 252-257 ("*Patt*")?

(B) Is claim 10 properly rejected under 35 U.S.C. § 103 as being unpatentable over *Patt* in view of *Hennessy*, "Computer Architecture: A Quantitative Approach", page 269 ("*Hennessy*")?

(C) Is claim 30 properly rejected under 35 U.S.C. § 103 as being unpatentable over *Patt* in view of *Hennessy*?

(D) Are claims 23, 39 and 40 properly rejected under 35 U.S.C. § 103 as being unpatentable over *Patt* in view of *McFarling*, "combining branch predictors"?

(E) Are claims 27 and 37 properly rejected under 35 U.S.C. § 103 as being unpatentable over *Patt*?

VII. GROUPING OF CLAIMS

Claims 10 and 30 form a first group.

Claims 21, 24 and 25 form a second group.

Claims 31 and 32 form a third group.

Claims 34, 35, 36, 38 and 39 form a fourth group.

Claims 23, 26 and 40 form a fifth group.

Claims 27 and 37 form a sixth group.

These groups are to be separately considered.

Claims 22, 28, 29 and 33 should be considered separately. The reasons why the claims of the respective groups and separately considered claims, if any, are separately patentable are found in the argument. 37 C.F.R. § 1.192 (c)(7).

VIII. ARGUMENT

(A) The rejection of claims 21, 22, 24-26, 28, 29, 31-36 and 38 under 35 U.S.C. § 102 as being anticipated by *Patt* is improper. The Appellant respectfully traverses the rejection of claims 21, 22, 24-26, 28, 29, 31-36 and 38 under 35 U.S.C. § 102.

Claim 21 is directed to branch prediction circuitry. The branch prediction circuitry includes a bimodal branch history table comprising a plurality of entries each for storing a prediction value and accessed by selected bits of a branch address, a fetch-based branch history table comprising a plurality of entries for storing a prediction value and accessed by a pointer generated from selected bits of the branch address and bits from a history register, each entry of the fetch-based branch history table operable for containing bits representing a prediction value for a plurality of branches in a fetch group, and a selector table comprising a plurality of entries each for storing a plurality of selector bits and accessed by a pointer generated from selected bits from the branch address and bits from the history register, each selector bit used for selecting between a bimodal prediction value accessed from the bimodal history table and a prediction value accessed from the fetch-based history table.

As an initial matter, the Examiner has pejoratively dismissed the Appellant's discussion of claim 21 in the Reply Under 37 C.F.R. § 1.111 mailed on August 22, 2002 ("Appellant's Reply") as being "pseudo arguments." (Paper No. 5, page 15.) The Appellant respectfully disagrees.

Moreover, the Examiner plainly mischaracterizes the Appellant's argument as simply stating the cited prior art does not teach the limitations. (Paper No. 5, page 15.) With respect to claim 21, for example, the Appellant plainly pointed out that at least, *Patt* does not disclose a fetch-based branch history table including entries operable for containing bits representing a prediction value for a plurality of branches in a fetch group. (Appellant's Reply, page 13.) The Examiner further states that the arguments "offer no support for their conclusionary statements." (Paper No. 5, page 15.) Of course, if the Examiner's arguments were correct, this would leave the Appellant in the illogical position of having to prove a negative. The Appellant plainly pointed out at least one difference between the prior art reference and the claimed invention. Furthermore, the Examiner's own particular assertions with respect to the limitations of claim 21, discussed hereinbelow, are themselves evidence supporting the Appellant's contention in the Appellant's Reply with respect to claim 21.

The Examiner asserts, with respect to the limitation in claim 21 that *Patt* teaches the fetched-based branch history table . . . in which each entry of the fetch-based branch history table is operable for containing bits representing a prediction value for a plurality of branches in a fetch group in § 3.1, subparagraph 4, page 253. (Paper No. 5, page 3.)[1] The teaching which purportedly discloses this limitation of claim 21, discloses in its entirety:

> gshare (*m*) – a modified version of the global variation of the Two-Level Adaptive Branch Predictor consisting of a single *m*-bit global branch history and a single pattern history table. The branch history and the branch address are XORed together to form the index into the pattern history table.

(*Patt*, § 3.1, subparagraph 4, page 253) (references omitted). Plainly there is no teaching therein directed to a fetch-based branch history table operable for containing bits representing a prediction value for a plurality of branches in a fetch group. In particular, there is nothing in this teaching that refers to a table containing bits representing a prediction value for a plurality of branches. This itself demonstrates that the Examiner has not been able to identify teaching in *Patt* that discloses the limitation in claim 1 with respect to a fetch-based branch history table. Moreover,

1 The Examiner refers to this as bracket 5. The brackets have been added to the reference by the Examiner and are not part of the original numbering scheme of *Patt*.

the Examiner, effectively, reads "fetch group" out of claim 21. The Examiner, without any objective evidence in support whatsoever, states that a fetch group is "nothing more than what is fetched." (Paper No. 5, page 3.)[2]

With respect to the limitations in claim 1 reciting a selector table comprising a plurality of entries, each for storing a plurality of selection bits... each selector bit used for selecting between..., the Examiner identifies disclosure in *Patt* directed to a two-level branch predictor selector. (Paper No. 5, page 3) (citing *Patt*, § 1 at bracket 1, page 252 and § 4.1, page 255). The teaching in *Patt* in the aforementioned disclosure with respect to the Branch Predictor Selection Table (BPST) discloses in its entirety:

> [t]he Branch Predictor Selection Table [] records which predictor was most frequently correct for the times this branch occurred with the associated branch history. This second level of history keeps track of the more accurate predictor for branches at different branch execution states.

(Patt, § 4.1, page 255.) *Patt* further teaches with respect to the BPST:

> [w]hen a branch is fetched, its instruction address in the current branch history is used to hash into the BPST. The associated counter is then used to select the appropriate prediction. By using the branch history to distinguish more execution states, 2-level predictor selection scheme can more accurately select the appropriate predictions.

(*Patt*, § 4.1, page 255.) Thus, there is nothing in the teaching relied upon disclosing the plurality of entries in the selector table, each for storing a plurality of selection bits.

As the Appellant has previously shown, *Patt* teaches a Branch Predictor Selection Table (BPST) having two-bit counters. (Applicant's Reply, page 11.) (citing *Patt*, Section 2, page 252.) The teaching in *Patt* with respect to such a BPST discloses hybrid branch prediction schemes in

---

2 While the Appellant acknowledges that claim terms may be given their broadest *reasonable* interpretation, the interpretation must be consistent with the specification, and the broadest reasonable interpretation must also be consistent with the interpretation that those skilled in the art would reach. MPEP § 2111. Note, for example, that in the detailed description in discussing a branch misprediction, a determination is made if the position of the mispredicted branch in a fetch group is at the end of the fetch group. (Detailed Description, page 26, lines 14-19.) Thus, interpreting the term "fetch group" to what is fetched makes no sense in view of the Specification.

which the two-bit counters in the BPST keep track of the currently more accurate predictor for a branch. (*Id.*) Additionally, the he two-level prediction scheme disclosed in *Patt* is a combination of the hybrid scheme with a first level history based on branch history registers (BHRs). (*Patt*, Section 4.1, page 255.) *Patt* explicitly teaches that the BPST used in the two-level prediction scheme is a table of two-bit counters just as disclosed in conjunction with the hybrid prediction scheme (*Patt*, Section 4.2, pages 255-56.) As discussed in conjunction therewith, as noted hereinabove, the two-bit counters of the BPST keep track of the predictor which is currently more accurate for a particular branch. These are incremented/decremented in accordance with which single-scheme predictor is correct. (*Patt*, Section 2, page 252.) Consequently, the two-bit counters as taught in *Patt* do not disclose a plurality of bits selector bits, each selector bit used for selecting between a bimodal prediction value… and a prediction value accessed from a fetch-based history table, as recited on claim 21. Indeed, *Patt* expressly teaches that the BPST provides a second level of history (record of which predictor was most frequently correct for the times a particular branch occurred with the associated branch history). (*Patt*, Section 4.1, page 255.) Thus, moreover, the two-bit counters in the BPST of *Patt* have nothing to do with a fetch group. This conclusion follows even assuming, for the sake of argument, that the Examiner's assertion that a fetch group is nothing more than what is fetched. (*See* Paper No. 5, page 3.)

Anticipation requires that a single prior art reference teach the identical invention of the claim. MPEP § 2131. That is, all the claim limitations, arranged as required by the claim, must be found in a single reference. *Id.* Therefore, because *Patt* has not been shown to teach the identical invention of claim 21, *Patt* does not anticipate claim 21. Consequently, the Appellant respectfully contends that claim 21 is allowable under 35 U.S.C. § 102 over *Patt*.

Claim 22 is directed to the branch prediction circuitry of Claim 21 and further including circuitry for updating the bimodal and fetch-based branch history tables. The circuitry is operable to set a corresponding entry in each of the bimodal and fetch-based branch history tables to a first value when a branch is taken at branch resolution time, and set a corresponding entry in each of the bimodal and fetch-based branch history tables to a second value when a branch is not taken at branch resolution time.

With respect to claim 22, the Examiner contends that *Patt* discloses the limitation with respect to the circuitry for updating the bimodal and fetch-based branch history tables and generally discussing branch predictor configurations, and single-scheme predictors. (Paper No. 5, page 3) (citing *Patt*, §§ 3 and 3.1, page 253.) No particular teaching directed to circuitry for updating the aforementioned branch history tables is identified. Also, the Appellant notes that § 3.1 is directed to single-scheme predictors. (*Patt*, § 3.1, page 253.) Moreover, as discussed in conjunction with claim 21, *Patt* has not been shown to teach, at least, the fetch-based branch history tables as recited therein. Because, for at least these reasons, *Patt* has not been shown to teach all of the limitations of claim 22, claim 22 is not anticipated by *Patt*. Therefore, the Appellant respectfully asserts that claim 22 is allowable under 35 U.S.C. § 102 over *Patt*.

Claim 24 is directed to the branch prediction circuitry of Claim 21 and further comprising circuitry for updating said selector table. The circuitry is operable to update a corresponding bit in a selected entry in said selector table with a first value when a bimodal prediction value from said bimodal branch history table correctly represents a corresponding branch resolution, and update a corresponding bit in a selected entry in said selector table with a second value when a fetch-based prediction value from said fetch-based branch history table correctly represents the corresponding branch resolution.

With respect to the limitation of claim 24, the Examiner relies on teaching in *Patt* that discloses a two-bit counter selection mechanism. (Paper No. 5, page 4) (citing *Patt* at bracket 3, page 252.) *Patt* also teaches that the counters are incremented or decremented depending on which single-scheme predictor is correct. (*Patt*, § 2, page 252.) Additionally, as previously discussed, *Patt* has not been shown to disclose a fetch-based branch history table as incorporated in claim 24 through its dependency on claim 21. Because *Patt* has not been shown to teach the identical invention of claim 24, the Appellant respectfully contends that claim 2 is allowable under 35 U.S.C. § 102 over *Patt*.

Claim 25 is directed to the branch prediction circuitry of Claim 21 wherein the plurality of selection bits are operable for selecting a first subset of prediction values from the bimodal branch history table and a second subset of prediction values from the fetch-based branch history table. Claim 25 incorporates by reference the limitations of claim 21, and therefore, *Patt* has not been

shown to teach all of the limitations of claim 25. Consequently, claim 25 is not anticipated by *Patt* and is, therefore, allowable under 35 U.S.C. § 102 over *Patt*

Claim 26 is directed to the branch prediction circuitry of Claim 23 wherein said circuitry for updating said selector table is further operable to maintain a value in a selected entry in said selector table when corresponding values from said bimodal and fetch-based branch history tables both correctly represent a corresponding branch resolution, and wherein said circuitry for updating said selector table is further operable to maintain a value in a selected entry in said selector table when neither values from said bimodal and fetch-based branch history tables correctly represent a corresponding branch resolution. As an initial matter, claim 23, from which claim 26 depends, has not been rejected as being anticipated by *Patt*. (Paper No. 5, page 2 and page 11.) Therefore, claim 26 is necessarily not anticipated by *Patt*. As discussed hereinbelow, the Appellant respectfully contends that claim 23 has not been demonstrated to be *prima facie* obvious. Consequently, claim 26 is also not *prima facie* obvious, and allowable under 35 U.S.C. § 103 over *Patt* and *McFarling*.

Claim 28 is directed to a processing system including a first branch history table and a second branch history table. The first branch history table comprises a plurality of bimodally accessed entries, each entry for storing a first set of branch prediction bits and the second branch history table comprises a plurality of fetch-based accessed entries each entry for storing a second set of branch prediction bits. The system also includes a selector for selecting, in response to a plurality of selection control bits, a set of prediction bits from a selected one of said sets of bits accessed from said first and second branch history tables, and a selector table comprising a plurality of entries, each entry for storing a plurality of selection control bits wherein the selection control bits are set as a function of a performance history of corresponding first and second sets of branch prediction bits stored in said first and second branch history tables.

With respect to the limitation in claim 28 directed to a second branch history table comprising a plurality of fetch-based accessed entries... , the Examiner relies on teaching directed to the gshare-scheme predictor discussed hereinabove with claim 21. (Paper No. 5, page 5) (citing *Patt*, Section 3.1, subparagraph 4, page 253). Similarly, with respect to the selector for selecting, in response to a plurality of selection of control bits... , the Examiner relies on teaching in *Patt* directed to the two-level branch predictor selection algorithm also discussed hereinabove in

conjunction with claim 21. (Paper No. 5, page 5) (citing *Patt*, Section 4.1, page 255 and Section 1, page 251 at bracket 1.) With respect to the selector table comprising a plurality of entries, each entry for storing a plurality of selection bits... , the Examiner relies on the same teaching in *Patt* drawn to the two-level branch predictor selection algorithm. (Paper No. 5, page 5) (citing *Patt*, Section 4.1, page 255 and page 252 at bracket 1.) As discussed hereinabove, the BPST includes two-bit counters for maintaining a second level of history. (*Patt*, Section 4.1, page 255.) Thus, the Appellant respectfully contends that *Patt* does not teach a selector for selecting in response to a plurality of selection control bits, a set of prediction bits from a selected one of the sets of bits accessed from the first and second branch history tables. Moreover, FIGURE 2, illustrating a two-input multiplexer for selecting between two predictions, of *Patt* suggests to the contrary.

Thus, for at least the aforesaid reasons, *Patt* has not been shown to teach the identical invention of claim 28. Consequently, claim 28 is allowable under 35 U.S.C. § 102 over *Patt*. MPEP § 2131.

Claim 29 is directed to the processing system of Claim 28 wherein said entries of said selector table are accessed using fetch-based accessing. Claim 29 has been rejected on teaching in *Patt* directed to a branch history register, and the hashing into the branch predictor selection table using a branch instruction address in the branch history. (Paper No. 5, page 6) (citing *Patt* Section 4.1, page 255). The Appellant respectfully disagrees that this would understood by one of ordinary skill in the art to teach a fetch-based accessing. As previously discussed, the Examiner has identified no teaching in *Patt* directed to the concept of a fetch group. It would be further understood by those of ordinary skill in the art that a fetch-based history vector is derived from the fetch group. (*See* e.g., Detailed Description, page 28, line 15 through page 32, line 5.) Moreover, claim 29 incorporates the limitations of claim 28, previously shown to be allowable under 35 U.S.C. § 102 over *Patt*. Thus, for at least the aforesaid reasons, the Appellant also respectfully contends that claim 29 is allowable under 35 U.S.C. § 102 over *Patt*.

Claim 31 is directed to the processing system of Claim 28 wherein said first and second branch history tables and said selector table form a portion of a branch execution unit. Claim 31 has been rejected on the disclosure in the Abstract of *Patt*. (Paper No. 5, page 6.) The abstract of *Patt* teaches in its entirety:

[v]ery accurate branch prediction is an important requirement for achieving high performance on deeply pipelined superscalar processors. To improve on the prediction accuracy of current single-scheme branch predictors, hybrid (multiple-scheme) branch predictors have been proposed. These predictors combine multiple single-scheme predictors into a single predictor. They use a selection mechanism to decide for each branch which single-scheme predictor to use. The performance of a hybrid predictor depends on a single-scheme predictor components and its selection mechanism. Using known single-scheme predictors and selection mechanisms, this paper identifies the most effective hybrid predictor mechanism. In addition, it introduces a new selection mechanism, the 2-level selector, which further improves the performance of the hybrid branch predictor.

(*Patt*, Abstract) (reference citations omitted). By the plain terms of the teaching relied upon, there is no disclosure indicating that the first and second branch history tables and selector tables form a portion of a branch execution unit. Anticipation requires that a single prior art reference teach the identical invention of the claim. MPEP § 2131. Furthermore, any reliance on inherency requires that a rationale or evidence be provided that shows that the missing descriptive matter is necessarily present in the thing described in the reference, and that it would be so recognized by persons of ordinary skill in the art. MPEP § 2112. Inherency may not be established by probabilities or possibilities. *Id*. Therefore, the Appellant respectfully contends that claim 31 has not been shown to be anticipated by *Patt*, and therefore is allowable under 35 U.S.C. § 102 over *Patt*.

Claim 32 is directed to the processing system of Claim 31 wherein said branch execution unit forms a part of a microprocessor. Claim 32 has also been rejected on the teaching in the abstract of *Patt*. (Paper No. 5, page 6.) While the Appellant does not deny that *Patt* discusses microprocessors in the abstract thereof, as previously discussed there is no mention of a branch execution unit. (*Patt*, Abstract, page 252.) Moreover, claim 32 depends from claim 31, and as previously discussed, *Patt* has not been demonstrated to teach the identical invention of claim 31. Therefore, for at least these reasons, claim 32 is also not anticipated by *Patt*. Thus, the Appellant respectfully contends that claim 32 is allowable under 35 U.S.C. § 102 over *Patt*.

Claim 33 depends from claim 32 and is directed to the system thereof and further comprising memory coupled to the microprocessor. The Examiner simply states that the branch selection predictors and tables themselves form "memory." (Paper No. 5, page 6.) The Appellant respectfully disagrees. The Appellant respectfully submits that persons of ordinary skill in the relevant art would

not understand memory coupled to a microprocessor to be the branch selection predictors and tables. Moreover, such an interpretation necessarily gives rise to a contorted meaning of "coupled" in view of the Examiner's contention that the branch selection predictors and tables themselves are part of the microprocessor. Plainly, *Patt* has not been shown to teach the invention of claim 33, and therefore claim 33 is allowable under 35 U.S.C. § 102 over *Patt*.

Claim 34 is directed to a method of performing branch predictions in a processing system including a bimodal branch history table, a fetch-based branch history table and a selector table. The method comprises accessing the bimodal branch history table to retrieve a first set of branch prediction bits, accessing the fetch-based branch history table to retrieve a set of second branch prediction bits, and selecting between the first and second sets of branch prediction bits in response to corresponding bits retrieved from the selector table. The sum of a number of bits in the first set of branch prediction bits and a number of bits in the second set of branch prediction bits is not less than a number of instructions in a fetch group. The method also includes updating the selector table as a function of actual branch resolution. Claim 34 has been rejected on the teaching in *Patt* directed to the 2-level branch predictor selection algorithm. (Paper No. 5, pages 6-7) (citing *Patt*, Section 4.1, page 255). The teaching relied upon has been discussed hereinabove in conjunction with, *inter alia*, claim 21. As discussed hereinabove, the Examiner has identified no teaching in *Patt* discussing a fetch group; the Examiner, without any objective evidence in support thereof, simply stated that a fetch group was nothing more than what is fetched. (Paper No. 5, page 3.) In rejecting claim 34, the Examiner similarly states that a fetch group in *Patt* might be defined as being no larger than one. (Paper No. 5, page 7.) The Appellant understands that to mean exactly one since it is illogical that no instructions be fetched. As an initial matter, the Appellant notes that defining a fetch group to be one instruction is inconsistent with the ordinary meaning of "group." Additionally, as previously discussed, interpreting a fetch group to mean the instruction that is fetched is also inconsistent with the Specification. Moreover, *Patt* would not be understood to teach selecting between the first and second sets of branch prediction bits in response to *corresponding bits* retrieved from the selector

table.[3] Consequently, the Appellant respectfully contends that *Patt* does not teach the identical invention of claim 34, and thus, is allowable under 35 U.S.C. § 102 over *Patt*.

Claim 35 is directed to the method of Claim 34 wherein said step of updating the selector table comprises determining if at least one of the first set of branch prediction bits correctly predicts the corresponding branch resolution outcome, and updating the corresponding entry in the selector table to a first logic value when the at least one of the first set of prediction bits correctly represents the branch resolution outcome. The method also includes determining if at least one of the second set of branch prediction bits correctly predicts the branch resolution outcome, and updating the corresponding entry in the selector table to a second logic value when the at least one of the second set of branch prediction bits correctly represents the branch resolution outcome. For at least the reason that claim 35 includes the limitations of claim 34 from which it depends, claim 35 is necessarily not anticipated by *Patt*. Therefore, claim 35 is also allowable under 35 U.S.C. § 102 over *Patt*.

Claim 36 is directed to the method of Claim 35 and further comprising determining if at least one bit of both the first and second sets of branch history bits correctly predict the branch resolution outcome, and maintaining the current value of corresponding bits in the corresponding selector table entry when the at least one bit of both the first and second sets of branch prediction bits correctly predict the branch resolution outcome. The method also includes determining if at least one bit of both the first and second sets of branch prediction bits incorrectly predict the branch resolution outcome, and maintaining the current value of corresponding bits in the corresponding selector table entry when the at least one bit of both the first and second sets of branch history bits incorrectly predict the branch history outcome. Claim 36 further depends from claim 35 and incorporates both the limitations of claim 35 and claim 34 by reference. For at least the reasons discussed hereinabove in conjunction with claims 34 and 35, the Appellant also respectfully asserts that claim 36 is allowable under 35 U.S.C. § 102 over *Patt*.

Claim 38 is directed to the method of claim 34 in which the step of accessing the fetch-based branch history table comprises the substep of generating an address from at least some bits of a

---

3 As the Appellant has noted, the branch predictor selection table in the two-level predictor selection mechanism actually includes two-bit counters. Nevertheless, selection between two inputs requires only a single selection state. This is consistent with the two-input multiplexer illustrated in FIGURE 2 of *Patt*. (*Patt*, FIGURE 2, page 255.)

branching instruction and bits retrieved from a history register. As discussed previously, *Patt* would not be understood to teach a fetch-based history table, therefore, *Patt* necessarily would not be understood to teach a fetch-based branch history table including generating an address from at least some bits of a branch instruction and bits retrieved from a history register. Moreover, claim 38 incorporates the limitations of claim 34 from which it depends, and therefore is necessarily not anticipated by *Patt*. Consequently, the Appellant respectfully asserts that claim 38 is allowable under 35 U.S.C. § 102 over *Patt*.

(B) Claim 10 has been improperly rejected under 35 U.S.C. § 103 as being unpatentable over *Patt* in view of *Hennessy*. The Appellant respectfully traverses the rejection of claim 10 under 35 U.S.C. § 103.

Claim 10 is directed to processing system including a first branch history table comprising a plurality of bimodally accessed entries for storing a first set of branch prediction bits, and a second branch history table comprising a plurality of fetch-based accessed entries for storing a second set of branch prediction bits. The system also includes a selector for selecting in response to a selection control bit selected from a set of selection control bits, a bit from a selected one of said sets of bits accessed from said first and second branch history tables. The selector table comprises a plurality of entries for storing said a set of selector bits as a function of a performance history of said first and second sets of branch prediction bits stored in said first and second branch history tables, wherein each entry in the tables comprises a 1-bit counter.

As previously discussed in conjunction with, *inter alia* claim 29, the Appellant respectfully contends that *Patt* would not be understood to teach fetch-based accessed entry. Additionally, the Examiner admits that *Patt* fails to teach one-bit counters. (Paper No. 5, pages 9-10.) The Examiner relies on Official Notice and an assertion that a one-bit implementation would be obvious to simplify the circuitry and thus become more efficient. (Paper No. 5, page 10.) Likewise the Examiner asserts that it would have been obvious to use one-bit counters as the simplest type of counters so as to reduce the circuit design for cost or space constraints. (*Id.*) These assertions have previously been made in Paper No. 3. (Paper No. 3, page 11.)

As the Appellant has previously shown, *Patt* cannot be modified to make the invention of claim 10. *Patt* teaches, in particular, a Branch Predictor Selection Table (BPST) having two-bit

counters. (Applicant's Reply, page 11) (citing *Patt*, Section 2, page 252.) The teaching in *Patt* with respect to such a BPST discloses hybrid branch prediction schemes in which the counters in the BPST keep track of the currently more accurate predictor for a branch. (*Id.*) The two-level prediction scheme disclosed in *Patt* is a combination of the hybrid scheme with a first level history based on branch history registers (BHRs). (Applicant's Reply, page 11) (citing *Patt*, Section 4.1, page 255.) *Patt* explicitly teaches that the BPST used in the two-level prediction scheme is a table of two-bit counters just as disclosed in conjunction with the hybrid prediction scheme (*Patt*, Section 4.2, pages 255-56.) As discussed in conjunction therewith, as noted hereinabove, the two-bit counters of the BPST keep track of the predictor which is currently more accurate for a particular branch. This is an essential element of the two-level prediction scheme of *Patt*; it provides the second level of history. (*See Patt*, Section 4.1, page 255.)

Consequently, modifying the two-bit counter BPST of *Patt* to use a one-bit counter changes the principle operation of *Patt*. Indeed, it renders it unsuitable for its intended purpose. Thus, there can be no motivation or suggestion to modify *Patt* to make the invention of claim 10, as a matter of law. MPEP § 2143.01 (citing *In re Gordon*, 733 F.2d 900, 221 U.S.P.Q. 1125 (Fed. Cir. 1984)). Furthermore, there must be some reasonable expectation of success in making the modification. MPEP § 2143.02. The reasonable expectation of success must be found in the prior art. MPEP § 2143. In the instant case, there can be no reasonable expectation of success because modifying *Patt* to replace the two-bit counters of the BPST reduces the second level of history in the two-level branch predictor scheme of *Patt*. This would be expected to adversely affect the figure of merit, that is, the misprediction rate in *Patt*.

The Examiner, in responding, admits that *Patt* requires the two-bit counters to perform correctly. (Paper No. 5, page 14.) The Examiner maintains, nonetheless, that the Examiner may "modify" the reference to make it simpler, even though that modification makes it unsuitable for its intended purpose. (*See* Paper No. 5, page 14.) Plainly, such a rationale is illogical. This implies that the Examiner, having learned from the instant Application that there are one-bit selector based branch prediction mechanisms, concludes that it would be obvious to "retool" *Patt* to make the claimed invention "to pursue a simpler version of the invention." It is by now well settled that such a rationale cannot sustain a *prima facie* showing of obviousness. *In re Rouffet*, 149 F.3d 1350, 1357,

47 U.S.P.Q.2d 1453, 1457-58 (Fed. Cir. 1998) (citations omitted). Such a rationale is also emblematic of the requirement that a motivation to modify the reference must be clear and particular. *In re Lee*, 277 F.3d 1338, 1343, 61 U.S.P.Q.2d 1430, 1433-34 (Fed. Cir. 2002); *In re Kotzab*, 217 F.3d 1365, 1371, 55 U.S.P.Q.2d 1313, 1317 (Fed. Cir. 2000); *In re Dembiczak*, 175 F.3d 994, 999, 50 U.S.P.Q.2d 1614, 1616 (Fed. Cir. 1999).

Because *Patt* has not been shown to teach or suggest all of the limitations of claim 10, and because there is no motivation to modify *Patt*, indeed *Patt* cannot be modified as a matter of law, to make the invention of claim 10, a *prima facie* showing of obviousness has not been made with respect to claim 10. Therefore, claim 10 is allowable under 35 U.S.C. § 103 over *Patt* and *Hennessy*.

(C) Claim 30 is improperly rejected under 35 U.S.C. § 103 as being unpatentable over *Patt* in view of *Hennessy*. The Appellant respectfully traverses the rejection of claim 30 under 35 U.S.C. § 103.

Claim 30 is directed to the processing system of claim 28 in which each said entry in the tables comprises a one-bit calendar. The Examiner has rejected claim 30 on the same basis as claim 10. (Paper No. 5, pages 10-11.) For the same reasons as discussed hereinabove in conjunction with claim 10, *Patt* is not modifiable to use one-bit counters. As a matter of law, there can be no motivation to modify *Patt* to make the invention of claim 30. Consequently, a *prima facie* showing of obviousness has not been made with respect to claim 30. Therefore, the Appellant respectfully contends that claim 30 is allowable under 35 U.S.C. § 103 over *Patt* and *Hennessy*.

(D) Claims 23, 39 and 40 are improperly rejected under 35 U.S.C. § 103 as being unpatentable over *Patt* in view of *McFarling*.

Claim 23 depends from claim 21 and recites the branch prediction circuitry thereof in which the history registry comprises a shift register and the branch prediction circuitry further comprises circuitry for updating the shift register by shifting in a preselected prediction value for each fetch group. The Examiner asserts that while *Patt* does not exclusively state the history register is comprised shift registers and the branch prediction circuitry further comprises circuitry for updating the shift register by shifting in a preselected prediction value, *McFarling* demonstrates that it was known at the time of the invention to implement such history registers as shift registers, and thus shift in a preselected prediction value. (Paper No. 5, page 11.) However, the Examiner does not

identify teaching in *McFarling* that updates the shift register by shifting in a preselected prediction value *for each fetch group*..

Indeed, the Examiner has not addressed this element of claim 23 whatsoever. (Paper No. 5, page 11.) A *prima facie* showing of obviousness requires, *inter alia*, that the references alone or in combination teach or suggest all the limitations of the claim. MPEP § 2143.03. Moreover, all words in the claim must be considered when judging the patentability thereof. *Id.* Consequently, on its face, the rejection of claim 23 fails to demonstrate that the references, alone or in combination teach or suggest all of the limitations of claim 23. Consequently, a *prima facie* showing of obviousness has not been made with respect to claim 23, and thus, claim 23 is allowable under 35 U.S.C. § 103 over *Patt* and *McFarling*.

Claim 39 is directed to the method of claim 38 in which the history register comprises a shift register. Claim 39 incorporates all of the limitations of claim 38 and the limitations of claim 34 from which claim 38 depends by reference. Consequently, for the reasons discussed hereinabove in conjunction with claim 34, the Appellant respectfully contends that neither *Patt* nor *Patt* in combination with *McFarling* teach or suggest all of the limitations of claim 39. Thus, claim 39 is allowable under 35 U.S.C. § 103 over *Patt* and *McFarling*.

Claim 40 depends from claim 39 and is directed to the method thereof in which the method further comprises updating the shift register by shifting in a prediction bit for each fetch group. Claim 40 has been rejected on the same basis as claim 23. (Paper No. 5, pages 11-12.) Again, the Examiner did not address the element in claim 40 directed to shifting in a prediction bit *for each fetch group*.. Consequently, a *prima facie* showing of obviousness has also not been made with respect to claim 40. Therefore, the Appellant respectfully asserts that claim 40 is allowable under 35 U.S.C. § 103 over *Patt* and *McFarling*.

(E) Claims 27 and 37 are improperly rejected under 35 U.S.C. § 103 as being unpatentable over *Patt*. The Appellant respectfully traverses the rejection of claims 27 and 37. Claim 27 is directed to the branch prediction circuitry of claim 23 in which the circuitry for updating the selector table is further operable to setting a value in a selected entry in the selector table to a value associated with the fetch-based table when corresponding values from the bi-modal and fetch-based branch history table do not both correctly predict a corresponding branch resolution outcome.

The Examiner admits that *Patt* fails to teach the limitation of claim 27. (Paper No. 5, page 13.) The Examiner then asserts that *Patt* demonstrates that it was known at the time of the invention to use two-level prediction. (Paper No. 5, page 13.) The Examiner then conclusively asserts that the two-level prediction is the fetch-based table, without any support identified in *Patt*. (Paper No. 5, page 13.) (Appellant notes that there is no reference in *Patt* to a "fetch-based" table.) The Examiner then asserts that it would have been obvious to one of ordinary skill in the art to implement the selector table with setting a value to the fetch-based table "as found in *Patt's* teaching." (Paper No. 5, page 13.) Again, the Appellant notes that the assertion with respect to a "fetch-based" table in *Patt* is a gloss on the teaching in *Patt*. Thus, the Appellant respectfully contends that *Patt* has not been shown to teach or suggest the limitation of claim 27.

With respect to a motivation, the Examiner asserts that it would have been obvious because one of ordinary skill would be motivated to have a prediction selector which increases its accuracy whenever possible. (Paper No. 5, page 13.) As an initial matter, the Examiner has not demonstrated that setting the selector bit as recited in claim 27 would improve the accuracy of *Patt* whatsoever. Furthermore, a motivation or suggestion to modify a reference must be based on objective evidence and found in one of three sources thereof: the teachings of the reference itself, the nature of the problem to be solved, or the knowledge of persons of ordinary skill in the art. MPEP § 2143.01. The Examiner has identified the motivation in none of these possible sources. Furthermore, the teachings with respect to a motivation must be clear and particular, and broad conclusory statements regarding the teachings standing alone are not evidence. *In re Lee*, 277 F.3d at 1343, 61 U.S.P.Q.2d at 1433-34 (Fed. Cir. 2002); *In re Kotzab*, 217 F.3d at 1370, 55 U.S.P.Q.2d at 1317 (Fed. Cir. 2000); *In re Dembiczak*, 175 F.3d at 199, U.S.P.Q.2d at 1616 (Fed. Cir. 1999). The factual question of motivation is material to patentability and cannot be resolved on subjective belief on unknown authority. *In re Lee*, 277 F.3d at 1343-44, 61 U.S.P.Q.2d at 1434. Indeed, it is illogical that a generic recitation that modifying the teaching of the references to improve the performance of the resulting system is a sufficient motivation for a *prima facie* showing of obviousness. If that were sufficient, then the requirement that there be some motivation or suggestion for combining references would simply disappear. It could always be said that the resulting modification leads to improved performance of the modified device. It is a rare circumstance indeed that an element would be incorporated in a claimed invention to diminish the performance thereof.

Thus, at least for the aforesaid reasons, the Appellant respectfully contends that a *prima facie* showing of obviousness has not been made with respect to claim 27. Consequently, claim 27 is allowable under 35 U.S.C. § 103 over *Patt*.

Claim 37 is directed to the method of Claim 35 and further including determining whether at least one bit of both the first and second sets of branch prediction bits correctly predict the branch resolution outcome, and maintaining the current value of corresponding bits in the corresponding selector table entry when at the least one bit of both the first and second sets of branch prediction bits correctly predict the branch resolution outcome. The method also includes updating the current selector table entry to a logic value associated with the fetch-based branch history table when neither of corresponding bits of the first and second sets of branch prediction bits correctly predicts the branch resolution outcome.

Claim 37 has been rejected on the same rationale as claim 27. (Paper No. 5, pages 13-14.) Therefore, for the reasons discussed in conjunction with claim 27, the Appellant also respectfully asserts that a *prima facie* showing of obviousness has not been made with respect to claim 37. Therefore, claim 37 is also allowable under 35 U.S.C. § 103 over *Patt*.

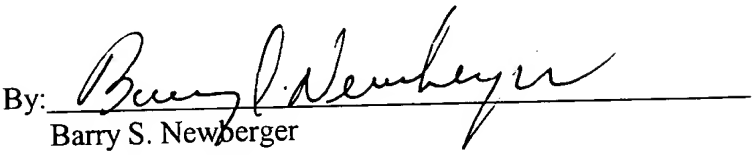
IX. CONCLUSION

For the reasons noted above, the rejection of claims 10 and 21-40 is in error. Reversal of the rejection and allowance of the Application is respectfully requested.

Respectfully submitted,

WINSTEAD SECHREST & MINICK P.C.

Attorneys for Appellant

By: 

Barry S. Newberger
Reg. No. 41,527

5400 Renaissance Tower
1201 Elm Street
Dallas, Texas 75270
(512) 370-2808

APPENDIX

10. A processing system comprising:

a first branch history table comprising a plurality of bimodally accessed entries for storing a first set of branch prediction bits;

a second branch history table comprising a plurality of fetch-based accessed entries for storing a second set of branch prediction bits;

a selector for selecting in response to a selection control bit selected from a set of selection control bits, a bit from a selected one of said sets of bits accessed from said first and second branch history tables; and

a selector table comprising a plurality of entries for storing said a set of selector bits as a function of a performance history of said first and second sets of branch prediction bits stored in said first and second branch history tables, wherein said each said entry in said tables comprises a 1-bit counter.

21. Branch prediction circuitry comprising:

a bimodal branch history table comprising a plurality of entries each for storing a prediction value and accessed by selected bits of a branch address;

a fetch-based branch history table comprising a plurality of entries for storing a prediction value and accessed by a pointer generated from selected bits of said branch address and bits from a history register, each entry of the fetch-based branch history table operable for containing bits representing a prediction value for a plurality of branches in a fetch group; and

a selector table comprising a plurality of entries each for storing a plurality of selection bits and accessed by a pointer generated from selected bits from said branch address and bits from said history register, each said selector bit used for selecting between a bimodal prediction value accessed from the bimodal history table and a prediction value accessed from said fetch-based history table.

22. The branch prediction circuitry of Claim 21 and further comprising circuitry for updating said bimodal and fetch-based branch history tables operable to:

set a corresponding entry in each of said bimodal and fetch-based branch history tables to a first value when a branch is taken at branch resolution time; and

set a corresponding entry in each of said bimodal and fetch-based branch history tables to a second value when a branch is not taken at branch resolution time.

23. The branch prediction circuitry of Claim 21 wherein said history register comprises a shift register and said branch prediction circuitry further comprises circuitry for updating said shift register by shifting in a preselected prediction value for each fetch group.

24. The branch prediction circuitry of Claim 21 and further comprising circuitry for updating said selector table operable to:

update a corresponding bit in a selected entry in said selector table with a first value when a bimodal prediction value from said bimodal branch history table correctly represents a corresponding branch resolution; and

update a corresponding bit in a selected entry in said selector table with a second value when a fetch-based prediction value from said fetch-based branch history table correctly represents the corresponding branch resolution.

25. The branch prediction circuitry of Claim 21 wherein the plurality of selection bits are operable for selecting a first subset of prediction values from the bimodal branch history table and a second subset of prediction values from the fetch-based branch history table.

26. The branch prediction circuitry of Claim 23 wherein said circuitry for updating said selector table is further operable to maintain a value in a selected entry in said selector table when corresponding values from said bimodal and fetch-based branch history tables both correctly represent a corresponding branch resolution, and wherein said circuitry for updating said selector table is further operable to maintain a value in a selected entry in said selector table when neither

values from said bimodal and fetch-based branch history tables correctly represent a corresponding branch resolution.

27. The branch prediction circuitry of Claim 23 wherein said circuitry for updating said selector table is further operable to set a value in a selected entry in said selector table to a value associated with said fetch-based table when corresponding values from said bimodal and fetch based branch history tables both do not correctly predict a corresponding branch resolution outcome.

28. A processing system comprising:

a first branch history table comprising a plurality of bimodally accessed entries, each entry for storing a first set of branch prediction bits;

a second branch history table comprising a plurality of fetch-based accessed entries each entry for storing a second set of branch prediction bits;

a selector for selecting, in response to a plurality of selection control bits, a set of prediction bits from a selected one of said sets of bits accessed from said first and second branch history tables; and

a selector table comprising a plurality of entries, each entry for storing a plurality of selection control bits wherein the selection control bits are set as a function of a performance history of corresponding first and second sets of branch prediction bits stored in said first and second branch history tables.

29. The processing system of Claim 28 wherein said entries of said selector table are accessed using fetch-based accessing.

30. The processing system of Claim 28 wherein said each said entry in said tables comprises a 1-bit counter.

31. The processing system of Claim 28 wherein said first and second branch history tables and said selector table form a portion of a branch execution unit.

32. The processing system of Claim 31 wherein said branch execution unit forms a part of a microprocessor.

33. The processing system of Claim 32 and further comprising memory coupled to said microprocessor.

34. A method of performing branch predictions in a processing system including a bimodal branch history table, a fetch-based branch history table and a selector table, the method comprising the substeps of:

accessing the bimodal branch history table to retrieve a first set of branch prediction bits;

accessing the fetch-based branch history table to retrieve a set of second branch prediction bits;

selecting between the first and second sets of branch prediction bits in response to corresponding bits retrieved from the selector table, wherein a sum of a number of bits in the first set of branch prediction bits and a number of bits in the second set of branch prediction bits is not less than a number of instructions in a fetch group; and

updating the selector table as a function of actual branch resolution.

35. The method of Claim 34 wherein said step of updating the selector table comprises the substeps of:

determining if at least one of the first set of branch prediction bits correctly predicts the corresponding branch resolution outcome;

updating the corresponding entry in the selector table to a first logic value when the at least one of the first set of prediction bits correctly represents the branch resolution outcome;

determining if at least one of the second set of branch prediction bits correctly predicts the branch resolution outcome; and

updating the corresponding entry in the selector table to a second logic value when the at least one of the second set of branch prediction bits correctly represents the branch resolution outcome.

36. The method of Claim 35 and further comprising the steps of:

determining if at least one bit of both the first and second sets of branch history bits correctly predict the branch resolution outcome;

maintaining the current value of corresponding bits in the corresponding selector table entry when the at least one bit of both the first and second sets of branch prediction bits correctly predict the branch resolution outcome;

determining if at least one bit of both the first and second sets of branch prediction bits incorrectly predict the branch resolution outcome; and

maintaining the current value of corresponding bits in the corresponding selector table entry when the at least one bit of both the first and second sets of branch history bits incorrectly predict the branch history outcome.

37. The method of Claim 35 and further comprising the steps of :

determining whether at least one bit of both the first and second sets of branch prediction bits correctly predict the branch resolution outcome;

maintaining the current value of corresponding bits in the corresponding selector table entry when at the least one bit of both the first and second sets of branch prediction bits correctly predict the branch resolution outcome; and

updating the current selector table entry to a logic value associated with the fetch-based branch history table when neither of corresponding bits of the first and second sets of branch prediction bits correctly predicts the branch resolution outcome.

38. The method of Claim 34 wherein said step of accessing the fetch-based branch history table comprises the substep of generating an address from at least some bits of a branching instruction and bits retrieved from a history register.

39. The method of Claim 38 wherein the history register comprises a shift register.

40. The method of Claim 39 wherein said method further comprises the steps of updating the shift register by shifting in a prediction bit for each fetch group.

AUSTIN_1\209880\1
7047-P295US 03/06/2003